มุมนักเล่น

DATABASE

ชัยยงค์ วงศ์ชัยสุวัฒน์

มาตรฐาน SQL1, SQL-89

เปิดเผยรายละเอียดมาตรฐาน SQL-89 ในส่วนของหลักไวยากรณ์ภาษาและคำหลักที่สงวนไว้เฉพาะ

ภาษาฐานข้อมูลมาตรฐาน SQL ที่ย่อมาจาก Structured Query Language เป็นภาษาเพื่อการจัดการและเรียกค้นข้อมูลที่เก็บในฐานข้อมูลเชิงสัมพันธ์ ซึ่งลักษณะของข้อมูลจะมีรูปแบบเป็นตารางของแถวและคอลัมน์ ภาษา SQL ไม่ใช่โปรแกรมจัดการฐานข้อมูลหรือเรียกกันย่อ ๆ ว่า DBMS ภาษา SQL ก็ไม่ใช่ผลิตภัณฑ์ที่แยกขายเดี่ยว ๆ เหมือนโปรแกรมคอมพิวเตอร์ทั่ว ๆ ไป แต่ภาษา SQL จะเป็นส่วนหนึ่งของโปรแกรมจัดการฐานข้อมูล โดยเป็นส่วนของภาษาและเครื่องมือเพื่อติดต่อกับโปรแกรมจัดการฐานข้อมูล ในการจัดการโครงสร้าง การจัดเก็บและการเรียกค้นข้อมูลในดิสค์

โปรแกรมจัดการฐานข้อมูลเชิงสัมพันธ์ทั้งในระดับ Desktop และระดับ Client/Server บนเครื่องพีซีและเวอร์กสเตชันต่างยอมรับมาตรฐานภาษา SQL มาเป็นเครื่องมือในการเรียกค้นและจัดการข้อมูลแทบทั้งสิ้น การบัญญัติภาษา SQL ให้เป็นมาตรฐานร่วมเพื่อให้บริษัทผู้ผลิตต่าง ๆ ยอมรับและนำไปใช้ภายในผลิตภัณฑ์ของตนเอง จะให้ผลดีต่อการใช้งานฐานข้อมูลในคอมพิวเตอร์มากหลายประการดังจะได้กล่าวถึงต่อไป

มาตรฐาน SQL ได้ผ่านขั้นตอนดำเนินการโดยหน่วยงานมาตรฐานสากลและหน่วยงานมาตรฐานระดับชาติ คือ ISO และ ANSI จนบัญญัติใช้เป็นทางการแล้วคือ SQL1 หรือ SQL-89 ได้มีการปรับปรุงแก้ไขเป็น SQL2 หรือ SQL-92 และกำลังมีการร่างมาตรฐาน SQL3 อยู่ ในขณะนี้ มุมดาต้าเบสฉบับนี้ขอนำเสนอมาตรฐาน SQL-89 ในส่วนของหลักไวยากรณ์ภาษาและคำหลักที่สงวนไว้เฉพาะ พร้อมกับประโยคคำสั่งเพิ่มเติมที่บริษัทผู้ผลิต DBMS ส่วนใหญ่บรรจุรวมไว้สูงกว่ามาตรฐาน SQL1 จนต้องปรับปรุงเป็น SQL2 ในเวลาต่อมา

ทบาทของภาษา SQL

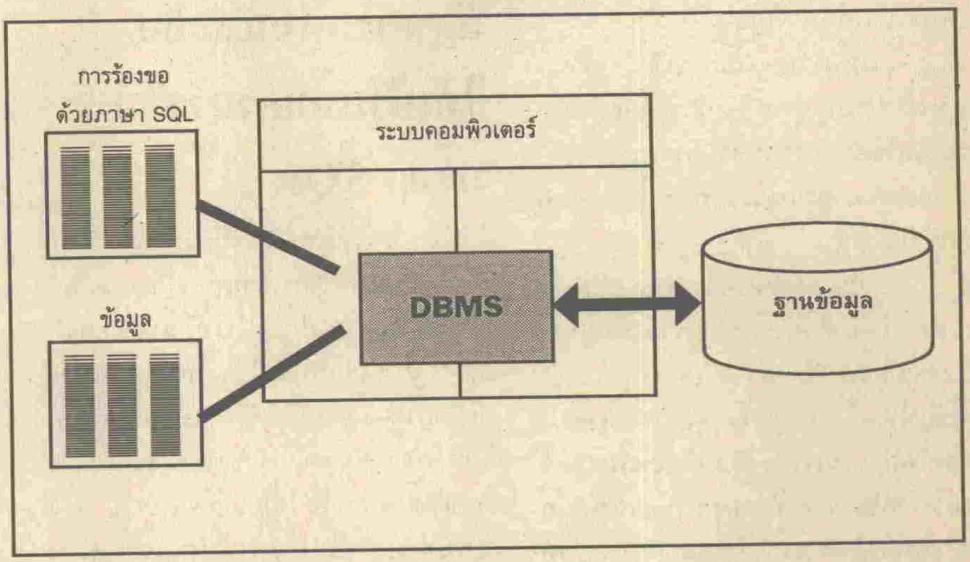
ภาษา SQL ได้รับการยอมรับให้ ามาใช้เป็นเครื่องมือเพื่อการเชื่อมโยง ใช้โปรแกรมคอมพิวเตอร์และระบบ อมพิวเตอร์ให้ติดต่อกับข้อมูลที่เก็บใน ักษณะเชิงสัมพันธ์ได้ดี (ดังรูปที่ 1) โดย ะใช้เป็นภาษาและเครื่องมือติดต่อกับ atabase Engine ซึ่งถือเป็นหัวใจสำคัญ องโปรแกรมจัดการฐานข้อมูล (DBMS) นการจัดโครงสร้างการจัดเก็บและการ รียกค้นข้อมูลในดิสค์

Database Engine จะรับการร้อง อด้วยภาษา SQL จากส่วนประกอบ ื่น ๆ ของ DBMS เช่น ส่วนของการสร้าง ฟอร์ม (Forms Facility) ส่วนสร้าง รายงาน (Report Writer) หรือส่วนเรียก ค้นโต้ตอบทันที (interactive query acility) จากโปรแกรมประยุกต์ที่เขียน จากผู้ใช้ และระบบคอมพิวเตอร์อื่น ๆ ดัง รูปที่ 2

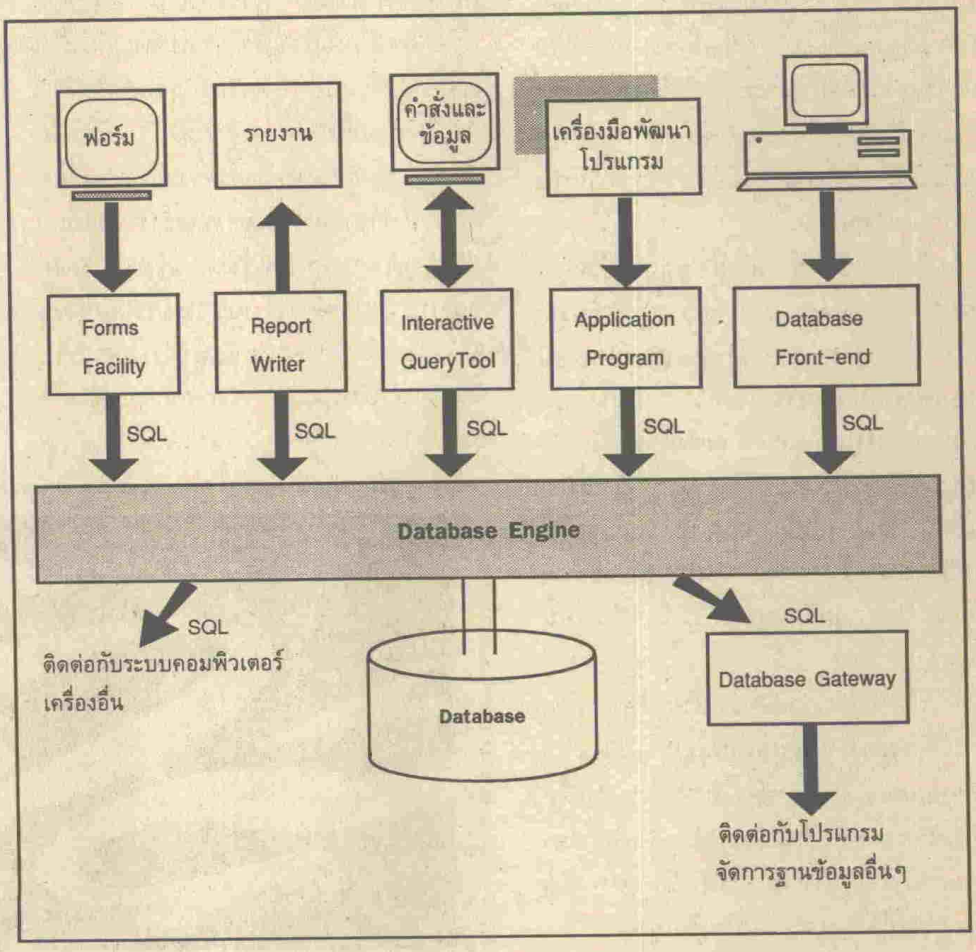
จากรูปที่ 2 จะเห็นว่าภาษา SQL แสดงบทบาททำงานหลายอย่าง ซึ่งอาจ แยกแยะบทบาทต่าง ๆ ดังนี้

- เป็นภาษาเรียกค้นโต้ตอบทันที โดยผู้ใช้เขียนคำสั่ง SQL ผ่านโปรแกรม ไปเรียกค้นข้อมูลทาง Database Engine ไปสู่ฐานข้อมูลในดิสค์ แล้วแสดงผลลัพธ์ ข้อมูลที่ส่งคืนกลับมาจากฐานข้อมูลที่หน้า จอทันที เป็นวิธีการเรียกค้นข้อมูลอย่าง ง่ายและสะดวก

- เป็นภาษาโปรแกรมฐานข้อมูล โดยผู้เขียนโปรแกรมประยุกต์สามารถ เขียนคำสั่ง SQL ฝังเข้าไปรวมในโปร- แกรมของภาษาอื่น ๆ เพื่อเรียกค้นข้อมูล ของฐานข้อมูล ซึ่งโปรแกรมที่ผู้ใช้งาน เขียนขึ้น หรือโปรแกรมยูทิลิตี้ของฐานข้อ- มูล เช่น การสร้างรายงานและการนำเข้า ข้อมูลที่เป็นรูปฟอร์มกรอกข้อมูล จะใช้วิธี



รูปที่ 1 การใช้ภาษา SQL ในการเรียกค้นฐานข้อมูล



รูปที่ 2 ส่วนประกอบสำคัญของ DBMS

นี้ในการเรียกค้นข้อมูล

- เป็นภาษาเพื่อการจัดการฐานข้อมูล โดยผู้ดูแลจัดการฐานข้อมูลสามารถใช้คำสั่ง SQL เพื่อกำหนดโครงสร้างและควบคุมการเรียกติดต่อข้อมูลที่จัดเก็บไว้

- เป็นภาษาของระบบ Client/Server โดยที่ระบบการคำนวณแบบ Client/Server ที่กำลังได้รับความนิยมนำมาใช้แทนระบบคำนวณรุ่นเก่าของเครื่องขนาดใหญ่ เมนเฟรมหรือมินิคอมพิวเตอร์ สามารถประยุกต์ใช้งานระบบฐานข้อมูลได้มากที่สุดในขณะนี้ โปรแกรมประยุกต์บนเครื่องพีซีนี้เป็นเครื่องผู้รับบริการหรือ Client จะใช้ภาษา SQL ในการติดต่อผ่านเครือข่าย LAN ไปยังเครื่อง Database Server เพื่อเรียกค้นข้อมูลร่วม ซึ่งสถาปัตยกรรมแบบ Client/Server ช่วยลดจราจรหรือปริมาณข้อมูลในเครือข่ายให้ลดน้อยลง โดยแบ่งแยกงานให้ทั้งเครื่องผู้รับบริการและผู้ให้บริการช่วยกันทำงานพร้อมกัน

- เป็นภาษาของฐานข้อมูลกระจาย โดยโปรแกรมจัดการฐานข้อมูลกระจาย (Distributed Database Management System) จะใช้ภาษา SQL ช่วยในการกระจายข้อมูลส่งไปยังระบบคอมพิวเตอร์ที่ต่ออยู่ในระบบ โปรแกรม DBMS ในแต่ละระบบจะใช้ภาษา SQL ในการติดต่อกับระบบอื่น ๆ ด้วยการส่งการร้องขอ SQL ไปเรียกค้นข้อมูล

- เป็นภาษาช่องประตูติดต่อฐานข้อมูลอื่น (database gateway language) โดยภาษา SQL จะเป็นสื่อหรือตัวกลางที่ใช้ในช่องประตูติดต่อ จาก DBMS ตระกูลหนึ่งไปยัง DBMS อีกตระกูลหนึ่งในการเรียกค้นข้อมูล

ลักษณะเด่นและข้อได้เปรียบของการใช้ภาษา SQL

จากหัวข้อที่แล้วจะเห็นว่าภาษา SQL นำไปใช้ได้ในด้านการจัดการฐานข้อมูลหลายอย่าง นับเป็นเครื่องมือมาตรฐานสำหรับเครื่องตั้งแต่เครื่องพีซี มินิคอมพิวเตอร์ เมนเฟรม และเครื่องเวอร์กสเตชันต่าง ๆ ซึ่งนอกจากการใช้งานหลายอย่างแล้ว ภาษา SQL ยังมีลักษณะเด่นอื่นที่มาสนับสนุนอีกหลายประการคือ

- ภาษา SQL ได้รับการยอมรับนำไปใช้ในผลิตภัณฑ์ DBMS ของแทบทุกบริษัท ดังนั้นถ้าการเรียนรู้ภาษา SQL ใน DBMS หนึ่งได้แล้ว การปรับเปลี่ยนไปสู่ DBMS ของอีกบริษัทหนึ่งจะทำได้ง่าย โดยไม่ต้องเรียนรู้จากศูนย์เครื่องมือช่วยงานฐานข้อมูลบนเครื่องพีซี เช่น การเรียกค้น การทำรายงาน และการสร้างโปรแกรมประยุกต์สามารถใช้กับฐานข้อมูล SQL ของหลายบริษัทได้ ผลของการใช้ภาษา SQL ทำให้ได้รับความนิยมมาก เพราะไม่ขึ้นกับบริษัทผู้ผลิต (Vendor Independeduce)

- บริษัทผู้ผลิตโปรแกรมจัดการฐานข้อมูลแบบ SQL หลายบริษัทจะมีผลิตภัณฑ์ตั้งแต่เครื่องระดับพีซี เวอร์กสเตชันในระบบเครือข่าย เครื่องมินิคอมพิวเตอร์ และเครื่องเมนเฟรม ครบทั้งหมด การโอนย้ายโปรแกรมจากระดับผู้ใช้เครื่องต่ำสุดไปสู่เครื่องสูงขึ้นถึงสุดสุดของเครื่องเมนเฟรมสามารถทำได้ง่าย ข้อมูลขององค์กรใหญ่สามารถย่อยและดาวน์โหลดมาสู่เครื่องระดับต่ำเป็นฐานข้อมูลแผนกหรือส่วนบุคคล ขณะเดียวกันเราสามารถสร้างโปรแกรมประยุกต์จากเครื่องเล็ก ๆ เพียงเป็นต้นแบบแล้วนำไปใช้เครื่องใหญ่ขึ้น แพงขึ้นของผู้ใช้หลายรายพร้อมกันได้ง่าย ภาษา SQL ช่วยในการโอนย้ายข้อมูลระหว่างคอมพิวเตอร์ระดับต่าง ๆ (Portabiltity)

- ภาษา SQL ได้รับการบัญญัติเป็นมาตรฐานโดยสถาบันมาตรฐานระหว่างประเทศ (ISO) และสถาบันมาตรฐานแห่งชาติอเมริกา (ANSI) ตั้งแต่ปี พ.ศ. 2529 และขยายเป็นมาตรฐาน SQL1 หรือ SQL-89 ในปี พ.ศ. 2532 เป็นมาตรฐาน SQL2 หรือ SQL-92 ในปี พ.ศ. 2535 ตามลำดับ นอกจากนี้ยังได้รับการยอมรับเป็นมาตรฐานด้วยการประมวลผลของสหรัฐอเมริกา (FIPS)

เป็นมาตรฐานโอนย้ายของบริษัท X/OPEN และกลุ่ม SQL Access Group หรือเรียกย่อ ๆ ว่า SAG ของกลุ่มบริษัทผู้ผลิตฐานข้อมูลและคอมพิวเตอร์ก็ได้กำหนดมาตรฐานระดับ call level สำหรับ SQL เพื่อการเชื่อมต่อฐานข้อมูลหลายแหล่ง ซึ่งก็เป็นต้นกำเนิดของข้อกำหนด ODBC (Open Database Connectivity) ของบริษัทไมโครซอฟต์ที่เคยกล่าวในมุมดาต้าเบสไปแล้ว

มาตรฐาน SQL1 หรือ SQL-89

ประโยคคำสั่ง SQL จะขึ้นต้นด้วยกริยาหรือ Verb ซึ่งจะเป็นคำหลัก (Keyword) อธิบายว่าคำสั่งใช้ทำงานอะไร เช่น การสร้าง (CREATE) การแทรก (INSERT) การลบ (DELETE) การเรียกค้นข้อมูล (SELECT) เป็นต้น ประโยคคำสั่งจะตามต่อด้วยหนึ่งหรือหลายอนุประโยค (clauses) ซึ่งอนุประโยคนี้จะบอกหรือระบุข้อมูลที่จะมากระทำตามกริยา หรือเป็นการบอกรายละเอียดเพิ่มเติมเกี่ยวกับการกระทำของกริยาของประโยคนั้น ๆ อนุประโยคต่าง ๆ จะขึ้นต้นด้วยคำหลัก เช่น WHERE, FROM, INTO, HAVING เป็นต้น ส่วนอนุประโยคนี้อาจเป็นส่วนบังคับต้องมีหรือเป็นส่วนเลือกอาจมีหรือไม่มีก็ได้ ทั้งนี้จะขึ้นกับกริยาของประโยคนั้น อนุประโยคจะประกอบด้วยชื่อตาราง หรือชื่อคอลัมน์ ซึ่งอาจมีคำหลักอื่นประกอบหรือเป็นค่าคงที่ หรือเป็นนิพจน์

ตัวอย่างเช่นในตารางที่ 1

รูปที่ 3 เป็นรายการคำหลักของมาตรฐาน SQL1 หรือ SQL-89 ที่ประกอบด้วยคำหลักที่เป็นคำกริยา (Verb) และคำหลักที่ใช้ในอนุประโยค ซึ่งคำหลักนี้เป็นคำสงวนจะใช้เป็นชื่อของวัตถุในฐานข้อมูล เช่น ชื่อตาราง ชื่อคอลัมน์ หรือชื่อผู้ใช้ต่าง ๆ อีกไม่ได้

ADA	CURRENT	GO	OF	SOME
ALL	CURSOR	GOTO	ON	SQL
AND	DEC	GRANT	OPEN	SQLCODE
ANY	DECIMAL	GROUP	OPTION	SQLERROR
AS	DECLARE	HAVING	OR	SUM
ASC	DEFAULT	IN	ORDER	TABLE
AUTHORIZATION	DELETE	INDICATOR	PASCAL	TO
AVG	DESC	INSERT	PLI	UNION
BEGIN	DISTINCT	INT	PRECISION	UNIQUE
BETWEEN	DOUBLE	INTEGAR	PRIMARY	UPDATE
BY	END	INTO	PRIVILEGES	USER
C	ESCAPE	IS	PROCEDURE	VALUES
CHAR	EXEC	KEY	PUBLIC	VIEW
CHARACTER	EXISTS	LANGUAGE	REAL	WHENEVER
CHECK	FETCH	LIKE	REFERENCES	WHERE
CLOSE	FLOAT	MAX	ROLLBACK	WITH
COBOL	FOR	MIN	SCHEMA	WORK
COMMIT	FOREIGN	MODULE	SECTION	
CONTINUE	FORTRAN	NOT	SELECT	
COUNT	FOUND	NULL	SET	
CREATE	FROM	NUMERIC	SMALLINT	

รูปที่ 3 คำหลักเฉพาะของภาษามาตรฐาน SQL-89

ตารางที่ 1

SELECT NAME, HIRE_DATE	; คำกริยา SELECT เป็นการคัดเลือกเรียกค้นเฉพาะข้อมูลของชื่อผู้แทนขายและวันเริ่มทำงาน
FROM SALESREPS	; อนุประโยคขึ้นต้นด้วย FROM ตามด้วยชื่อตารางผู้แทนจำหน่าย
WHERE SALES>500000	; อนุประโยคขึ้นต้นด้วย WHERE ตามด้วยนิพจน์ตรรกระบุยอดขายสูงถึง 500000 โดยคำขีดเส้นใต้เป็นคำหลักหรือ Keyword

คำสั่งใน SQL1 หรือ SQL-89

มาตรฐานภาษา SQL1 หรือ SQL 89 อาจแบ่งประโยคคำสั่งได้เป็น 3 กลุ่มคือ

1. ภาษานิยามข้อมูล (Data Definition Language-DDL) ได้แก่ ประโยคคำสั่งเพื่อการกำหนดโครงสร้างฐานข้อมูล กฎ ความถูกต้องของข้อมูล และสิทธิของผู้ใช้งาน กลุ่มนี้มีคำสั่งต่าง ๆ ดังนี้

- CREATE SCHEMA โดย Schema หนึ่งจะประกอบด้วยตารางที่มองจากมุมของผู้ใช้ (UserView) และสิทธิ์ของผู้ใช้ชุดหนึ่ง ผู้ใช้รายหนึ่งจะมีได้

เพียง Schema เดียวเพราะในฐานข้อมูลหนึ่งมีได้หลาย Schema

- CREATE TABLE การสร้างตารางข้อมูล
- CREATE VIEW การสร้างตารางที่มองจากผู้ใช้
- GRANT การกำหนดสิทธิ์ในการใช้ข้อมูล

2. ภาษาจัดการข้อมูล (Data Manipulation Language-DML) ได้แก่ ประโยคคำสั่งที่ใช้เพื่อการเรียกค้นและดัดแปลงแก้ไขข้อมูลต่าง ๆ ซึ่งอาจแบ่งย่อยเป็นสองกลุ่ม กลุ่มแรกเรียก Cursor dependent โดยมีกลไกเรียก Cursor ให้มีการเรียกอ่านผลลัพธ์จากกลุ่มหรือเซตที่ค้นพบคราวละแถวหรือเรคอร์ด ประโยคคำสั่งกลุ่มนี้จะต้องมีชื่อ Cursor ปรากฏอยู่ด้วย คำสั่งในกลุ่มย่อยมีดังนี้

- DECLARE <ชื่อ Cursor> การกำหนดชื่อของ Cursor
- OPEN <ชื่อ Cursor> การเปิด Cursor
- FETCH <ชื่อ Cursor> การเรียกข้อมูลแถวใหม่ตามชื่อ Cursor
- CLOSE <ชื่อ Cursor> การปิด Cursor
- DELETE <ชื่อ Cursor> การลบข้อมูลที่ Cursor
- UPDATE <ชื่อ Cursor> การดัดแปลงแก้ไขข้อมูลที่ Cursor

ส่วนกลุ่มย่อยหลังจะเรียกว่า Cursor Independent ที่ไม่มีกลไกของ Cursor เข้ามาเกี่ยวข้อง มีการเรียกแสดงผลตามกลุ่ม หรือเซตที่ค้นพบ คำสั่งในกลุ่มนี้ตามมาตรฐาน SQL ประกอบด้วย

- SELECT การเลือกเรียกค้นข้อมูล
- INSERT การแทรกแถวของเรคอร์ด
- UPDATE การแก้ไขข้อมูล
- DELETE การลบแถวของเรคอร์ด
- COMMIT การยกเลิกงาน

รูปที่ 4 ประโยคคำสั่งตามมาตรฐาน SQL1 หรือ SQL-89

```
CREATE SCHEMA AUTHORIZATION author
CREATE TABLE table (table-def-item-list)
CREATE VIEW view [(coumn-list)]
   AS query-spec
   [WITH CHECK OPTION]
GRANT {ALL[PRIVILEGES]|privilege-list}
   ON table
   TO {PUBLIC | user-list}
   [WITH GRANT OPTION]
```

ก.ประโยคนิยามข้อมูล

```
DECLARE cursor CURSOR FOR query-expr [ORDER BY sort-item list]
OPEN cursor
CLOSE cursor
FETCH cursor INTO host-variable-list
DELETE FROM table WHERE CURRENT OF cusor
UPDATE table SET assignment-list WHERE CURRENT OF cursor
```

*ข.ประโยคจัดการข้อมูลส่วน **Cursor dependent***

```
MODULE module LANGUAGE language
AUTHORIZATION author
PROCEDURE procedure parameter-declaration
```

ง.ภาษาโมดูล

```
SELECT[ALL | DISTINCT] {select-item-list|*}
   INTO host-variable-list
   FROM table-ref-list
   [WHERE search-condition]
INSERT INTO table [(column-list)]
   VALUES {(insert-item-list) | query-spec]
DELETE FROM table [WHERE search-condition]
UPDATE table SET assignment-list [WHERE search-condition]
COMMIT WORK
ROLLBACK WORK
```

*ค.ประโยคจัดการข้อมูลส่วน **Cursor Independent***

```
query-expr:    query-item | query-expr UNION [ALL] query-item
query-item:    query-spec | (query-expr)
query-spec:    SELECT [ALL | DISTINCT] {select-item-list |*}
                  FROM table-ref-list
                  [WHERE search-condition]
                  [GROUP BY colume-ref-list]
                  [HAVING search-condition]
subquery:      (SELECT [ALL | DISTINCT] {select-item |*}
                  FROM table-ref-list
                  [WHERE search-condition]
                  [GROUP BY column-ref-list]
                  [HAVING search-condition])
```

จ. นิพจน์การเรียกค้น

- ROLLBACK การเรียกคืนงานเดิมที่มีไว้ก่อน

3. ภาษาโมดูล (Module Language) ได้แก่ ประโยคคำสั่งของการนำคำสั่ง DML มารวมเป็นโปรแกรมย่อย และการรวมคำสั่ง DML และโปรแกรมย่อยต่าง ๆ มารวมเป็นโมดูล แสดงเป็นประโยคคำสั่งจัดการข้อมูลของภาษา SQL เพื่อให้สามารถบรรจุรวมเป็นโมดูลหรือรวมในโปรแกรมของภาษาระดับสูงอื่น คำสั่งประกอบด้วย

- MODULE เพื่อรวมโปรแกรมย่อยต่าง ๆ ให้เป็นโมดูล

- PROCEDURE โปรแกรมย่อยจะประกอบด้วยคำสั่งกลุ่ม DML หลาย ๆ คำสั่ง

ตามมาตรฐาน ANSI/ISO SQL1 มีการกำหนดไวยากรณ์ภาษา SQL เป็นสัญลักษณ์แบบ Backus Normal Form เขียนย่อ ๆ ว่า BNF ซึ่งยากต่อการอ่านหรือเข้าใจ เพราะในมาตรฐานจะเป็นการแสดงประโยคคำสั่งแบบล่างขึ้นบน (bottom up) แทนที่จะเป็นแบบบนลงล่าง (top-down) ที่จะให้ภาพได้ดีกว่า นอกจากนี้ในมาตรฐานเดิมยังใช้ศัพท์ของเทอมที่ไม่ค่อยคุ้นเคยกัน ในที่นี้จึงขอแสดงประโยคคำสั่งที่เข้าใจได้ง่ายขึ้น ดังรูปที่ 4 โดยในประโยคคำสั่งเหล่านี้ จะใช้สัญนิยมดังต่อไปนี้

ก. คำหลักเฉพาะของ SQL จะแสดงเป็นตัวพิมพ์ใหญ่

รูปที่ 4 (ต่อ)

search-condition:	Search-item | search-item {AND | OR} search-item
search-item:	[NOT] {search-test | (search-condition)}
search-test:	comparison-test | between-test | like-test | null-test | set-test | quantified-test | existence-test
comparison-test:	expr {= | <> | < | <=| > | >=}{expr | subqurey}
between-test:	expr [NOT] BETWEEN expr AND expr
like-test:	column-ref [NOT] LIKE value [ESCAPE value]
null-test:	column-ref IS [NOT] NULL
set-test:	expr [NOT] IN {value-list | subquery}
quantified-test:	expr {= | <> | < | <=|> | >=} [ALL | ANY | SOME] subquery
existence-test:	EXISTS subquery

ฉ.เงื่อนไขการค้นหา

expr:	expr-item | expr-item {+|-|*|/} expr-item
expr-item:	[+ |-] [value | column-ref | function | (expr)}
value:	literal | USER | host-variable
host-variable:	variable [[INDICATOR]variable]
function:	COUNT (*) | distinct-fcn | all-fcn
distinct-fonction:	{AVG | MAX | MIN | SUM | COUNT} (DISTINCT column-ref)
all-function:	{AVG | MAX | MIN | SUM | COUNT} ([ALL]expr)

ช.นิพจน์ทั่วไป

assignment:	column=[expr | NULL]
sort-item:	{column-ref | integer} [ASC | DESC]
insert-item:	{value | NULL}
select-item:	expr
table-ref:	table [table-alias]
column-ref:	[{table | alias} .] column
table-def-item:	column-def | UNIQUE (column-list)| PRIMARY KEY (column-list)| FOREIGN KEY (column-list) REFERENCES table [(column-list)]
column-def:	column data-type [DEFAULT {literal | USER | NULL}]
column-constraint:	[column-constraint] NOT NULL [UNIQUE | PRIMARY KEY]| CHECK (search-condition)| REFERENCES table [(column-list)]
privilege:	SELECT | INSERT | DELETE | UPDATE [column-list)]

ซ. ส่วนประกอบที่พบในประโยคคำสั่ง

table:	ชื่อตาราง
column:	ชื่อคอลัมน์
user:	ชื่อผู้ใช้ฐานข้อมูล
variable:	ชื่อตัวแปร
literal:	ตัวเลขหรือตัวอักษรที่บรรจุภายในเครื่องหมายคำพูด
integer:	ค่าเลขจำนวนเต็ม
data-type:	ชนิดข้อมูลใน SQL
alias:	ชื่อเล่นใน SQL
cursor:	ชื่อ cursor ใน SQL
author:	ชื่อเจ้าของ
module:	ชื่อ module
lanaguge:	ชื่อภาษาโปรแกรม
precedure:	ชื่อ precedure

ฌ. ชื่อหรือค่าคงที่ที่พบในประโยคคำสั่ง

ข. องค์ประกอบสำคัญที่ต้องมีรวมในไวยากรณ์ เช่น ชื่อตาราง ชื่อตารางตามมาตรฐานของผู้ใช้ (user view) จะเขียนด้วยตัวเล็ก หรืออาจอยู่ในวงเล็บสัญลักษณ์น้อยกว่าและมากกว่า เช่น <ชื่อตาราง>

ค. รายการประกอบ (element list) จะหมายถึง ส่วนประกอบหรือรายการประกอบหลายตัวที่แยกจากกันด้วยเครื่องหมายคอมมา (Comma)

ง. สัญลักษณ์เส้นในแนวดิ่ง (Verical bar) เขียนเป็น | หมายถึงการเลือกในระหว่างองค์ประกอบทางไวยากรณ์ที่มีให้เลือกหลายส่วนเพียงหนึ่งเดียว

จ. สัญลักษณ์วงเล็บเหลี่ยม [ ] หมายถึงส่วนเลือกที่อาจมีหรือไม่มีก็ได้

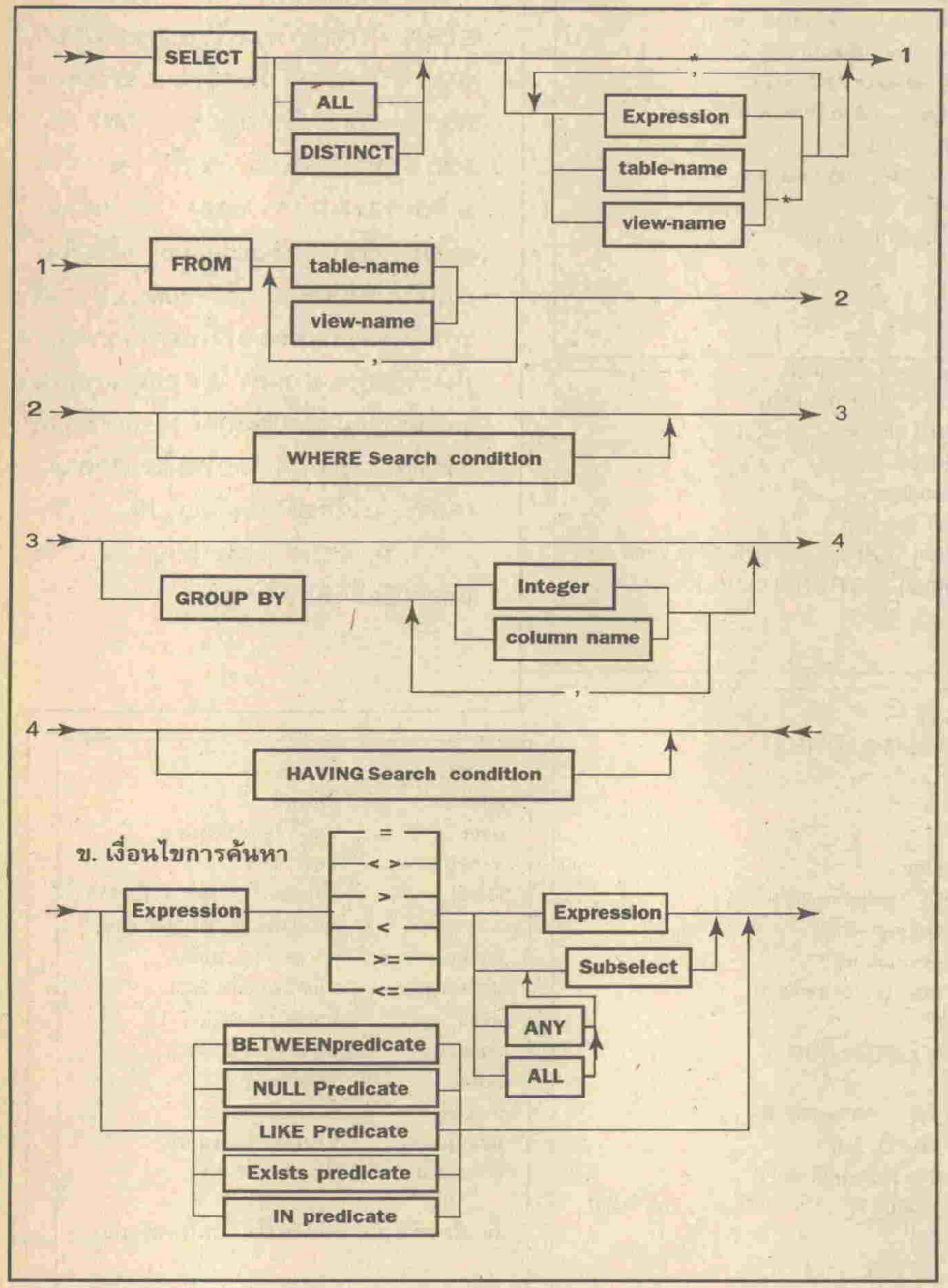
ฉ. สัญลักษณ์วงเล็บปีกกา หมายถึงส่วนเลือกที่ต้องเลือกมาแน่นอนจากรายการองค์ประกอบทางไวยากรณ์

ตัวอย่างเช่น

```
SELECT [ALL / DISTINCT] <select-item-list>/*
        INTO <host-variable-list>
        FROM <table-ref-list>
        [WHERE <search-condition>]
```

ประโยคคำสั่งนี้มี SELECT, INTO,FROM, WHERE, ALL, DISTINCT เป็นคำหลักเฉพาะ หลังคำหลัก SELECT จะต้องตามด้วยเครื่องหมายดอกจัน หรือ select-item-list อย่างแน่นอน เพราะอยู่ภายในวงเล็บปีกกา โดยผู้ใช้สามารถเลือกอย่างใดอย่างหนึ่งตามที่กั้นแบ่งเส้นตรงในแนวดิ่ง หลังคำหลัก SELECT และก่อนหน้าวงเล็บปีกกา อาจมีคำหลัก ALL หรือ DISTINCT ก็ได้ เพราะบรรจุอยู่ในวงเล็บเหลี่ยมของการเลือก อาจมีหรือไม่มีก็ได้ เช่นเดียวกับส่วนประกอบ WHERE search condition จะมีหรือไม่มีก็ได้ รายการ list เช่น ชื่อตาราง (table-ref-list) ถ้ามีหลายตาราง ก็จะเป็นชื่อตารางแต่ละตารางแล้วกั้นด้วย Comma

ประโยคคำสั่งในรูปที่ 4 เป็นแบบย่อให้สั้นเพื่อความเข้าใจง่าย และได้ตัดรายละเอียดของประโยค CREATE SCHEMA และประโยคประเภทโมดูล เพราะไม่ค่อยนิยมใช้กันในปัจจุบัน ซึ่งคำสั่งต่าง ๆ เหล่านี้เราอาจเขียนเป็นผังไวยากรณ์ที่มองดูได้ง่ายยิ่งขึ้นอีก ดังรูปที่ 5



รูปที่ 5 ตัวอย่างผังไวยากรณ์คำสั่ง SELECT และเงื่อนไขการค้นหา

ประโยคคำสั่งเพิ่มขยายและยอมรับในหมู่ผู้ผลิตต่าง ๆ

คำสั่งต่าง ๆ ในมาตรฐาน SQL1 หรือ SQL-89 เป็นเพียงส่วนหนึ่งที่ผลักดันให้เกิดมาตรฐานของภาษาที่เป็นส่วนกลางขึ้น ดังนั้นจึงยังไม่สมบูรณ์ต่อการใช้งานฐานข้อมูลทั่วไป บริษัทผู้ผลิตโปรแกรมจัดการฐานข้อมูล (DBMS) ต่าง ๆ ที่มีขายในท้องตลาดจึงเพิ่มขยายคำสั่งเพิ่มขึ้นไปอีก ได้แก่คำสั่ง

- ALTER TABLE การแก้ไขในตาราง
- COMMENT การแทรกข้อความอธิบาย
- CREATE INDEX การสร้างดัชนีหรืออินเด็กซ์
- CREATE SYNONYM การสร้างชื่อพ้อง
- DROP INDEX การยกเลิกดัชนี หรืออินเด็กซ์
- DROP SYNONYM การยกเลิกชื่อพ้อง
- DROP TABLE การยกเลิกตาราง
- DROP VIEW การยกเลิกตารางจากผู้ใช้
- LOCK การป้องกันตารางข้อมูล
- REVOKE การถอนสิทธิ์ของผู้ใช้
- LABEL การกำหนดไตเติ้ลของตารางหรือคอลัมน์
- EXPLAIN การอธิบายแผนการเรียกค้นข้อมูล
- PREPARE การเตรียมประโยค SQL เพื่อการทำงานตามอย่าง dynamic
- EXECUTE การทำงานตามประโยคคำสั่ง SQL อย่าง dynamic
- DESCRIBE การอธิบายการเรียกค้นที่ตระเตรียมด้วยคำสั่ง PREPARE

ประโยคคำสั่งที่เพิ่มขยายนี้เป็นตัวผลักดันต้องปรับปรุงมาตรฐาน SQL จนเกิดมาตรฐาน SQL2 หรือ SQL-92 ซึ่งมุมดาต้าเบสจะนำเสนอมาตรฐาน SQL-92 ในฉบับต่อไป โดยที่มาตรฐานใหม่ SQL-92 เริ่มมองอนาคตมากขึ้น จึงได้วางหลักไวยากรณ์ไว้ละเอียดยิ่งขึ้นกว่า SQL-89 มาก ในขณะนี้ซึ่งเป็น พ.ศ. 2537 หรือ ค.ศ. 1994 หลังจากที่มาตรฐาน SQL-92 ได้ออกมาแล้วเกือบครบสองปี ก็ยังไม่ได้มีผลิตภัณฑ์ของบริษัท DBMS ใดสามารถให้ภาษา SQL ของตนเดินตามมาตรฐาน SQL-92 ได้ครบถ้วน ผลิตภัณฑ์ SQL Server ของบริษัทไมโครซอฟต์เองก็ได้ตั้งเป้าให้ผลิตภัณฑ์ของตนเป็นไปตามมาตรฐาน SQL-92 ให้ปี 1996 เลยทีเดียว

ขณะนี้องค์กรมาตรฐานสากลของ ISO และ ANSI ต่างก็กำลังประสานงานร่างมาตรฐาน SQL3 อยู่ โดยเน้นให้เป็นระบบแบบ Intelligent Database ของการรวมรูปแบบวัตถุเชิงวิสัย (Object Orientation) เข้ากับระบบฐานความรอบรู้ (Knowledge Based System) ของศาสตร์ปัญญาประดิษฐ์

ตารางที่ 2

CHARACTER (len)	ข้อมูลตัวอักษรแบบความยาวจำกัดจำนวน len ตัว
CHAR (len)	ข้อมูลตัวอักษรแบบความยาวจำกัดจำนวน len ตัว
NUMERIC (precision, scale)	ข้อมูลตัวเลขทศนิยมระบบความแม่นยำ (precision) และหลักจุดทศนิยม (scale)
DECIMAL (precision, scale)	ข้อมูลตัวเลขทศนิยมระบบความแม่นยำ (precision) และหลักจุดทศนิยม (scale)
DEC (precision, scale)	ข้อมูลตัวเลขทศนิยมระบบความแม่นยำ (precision) และหลักจุดทศนิยม (scale)
INTEGER	ข้อมูลตัวเลขจำนวนเต็ม
INT	ข้อมูลตัวเลขจำนวนเต็ม
SMALLINT	ข้อมูลตัวเลขจำนวนเต็ม
FLOAT (precision)	ข้อมูลตัวเลขทศนิยมระบุความแม่นยำ (precision)
REAL	ข้อมูลตัวเลขทศนิยมระบุความแม่นยำต่ำ
DOUBLE PRECISION	ข้อมูลตัวเลขทศนิยมระบุความแม่นยำสูง

ประเภทข้อมูลใน SQL1

ในมาตรฐาน SQL1 ได้กำหนดประเภทข้อมูลไว้ดังในตารางที่ 2

สรุป

มาตรฐานภาษา SQL1 หรือ SQL-89 เป็นหลักไวยากรณ์ภาษาสำหรับการจัดการและเรียกค้นฐานข้อมูลเชิงสัมพันธ์ร่วม ที่บัญญัติโดยสถาบันมาตรฐานสากลของ ISO/ANSI เป็นที่ยอมรับใช้กันมากในปัจจุบัน มาตรฐาน SQL-89 ยังไม่สมบูรณ์ในตัวเต็มที่ จึงมีการเพิ่มขยายคำสั่งเกิดขึ้นในผลิตภัณฑ์ DBMS ของบริษัทต่าง ๆ มากมาย หลักไวยากรณ์ภาษา SQL ในขณะนี้จึงแตกต่างกันบ้างในผลิตภัณฑ์ของแต่ละบริษัท แต่หลักการใหญ่ ๆ จะคล้ายคลึงกัน การเรียนรู้และเข้าใจในภาษา SQL จะช่วยให้เราใช้งานฐานข้อมูลในอนาคตได้เป็นอย่างดี ภาษานี้ไม่ยากขอเพียงให้ตั้งใจจริงเท่านั้น